CLAIMS

1. A method of synchronizing objects between first and second object stores, comprising:

maintaining a reference store containing identifying data segments corresponding respectively to a plurality of objects that are to be synchronized between the first and second object stores;

submitting a particular identifying data segment from the reference store to a first interface associated with an application program that maintains the first object store; and

querying a second interface associated with the application program to determine whether the object corresponding to said particular identifying data segment has been changed in the first object store.

2. One or more computer-readable media comprising computer executable instructions for performing the method recited in claim 1.

3. A method as recited in claim 1, further comprising updating the second object store with the changed object from the first object store if the object has been changed in the first object store.

4. A method as recited in claim 1, further comprising:

determining whether the object corresponding to said particular identifying data segment has been changed in the second object store; and

updating the first object store with the changed object from the second object store if the object has been changed in the second object store.

5. A method as recited in claim 1, further comprising receiving the identifying data segments from one or more interfaces associated with the application program, wherein each identifying data segment contains an object identifier for a corresponding object in the first object store.

6. A method as recited in claim 1, further comprising receiving the identifying data segments from one or more interfaces associated with the application program, wherein some of the identifying data segments contain data representing properties that are supported by the second object store but that are not supported by the first object store.

7. A method of synchronizing objects between first and second object stores, comprising:

maintaining a reference store containing identifying data segments corresponding respectively to a plurality of objects that have previously been synchronized between the first and second object stores;

submitting a particular identifying data segment from the reference store to a first interface method associated with an application program that maintains the first object store;

in response to submitting said particular identifying data segment from the reference store, receiving a handle that corresponds to said particular identifying data segment; and

querying a second interface associated with the application program with the received handle to determine if the object corresponding to said particular identifying data segment has been changed in the first object store since the last synchronization.

8. One or more computer-readable media comprising computer executable instructions for performing the method recited in claim 7.

9. A method as recited in claim 7, further comprising if the object has been changed in the first object store, updating the second object store with the changed object from the first object store.

10. A method as recited in claim 7, further comprising:

maintaining a mapping between identifying data segments and object identifiers that are used in the second store to identify objects; and

submitting the object identifier that is mapped to the particular identifying data segment to a program that maintains the second object store to determine whether the object corresponding to said particular identifying data segment has been changed in the second object store.

11. A method as recited in claim 7, further comprising:

maintaining a mapping between identifying data segments and object identifiers that are used in the second store to identify objects;

submitting the object identifier that is mapped to the particular identifying data segment to a program that maintains the second object store to determine whether the object corresponding to said particular identifying data segment has been changed in the second object store; and

if the object has been changed in the second object store, updating the first object store with the changed object from the second object store.

12. A method as recited in claim 7, further comprising receiving the identifying data segments from one or more interfaces associated with the application program, wherein each identifying data segment contains an object identifier for a corresponding object in the object store.

13. A method as recited in claim 7, further comprising receiving the identifying data segments from one or more interfaces associated with the application program, wherein some of the identifying data segments contain data representing properties that are supported by the second object store but that are not supported by the first object store.

14. One or more computer-readable media comprising computer executable instructions that, when executed, direct a computing system to synchronize objects between first and second object stores, the instructions comprising:

maintaining a reference store containing identifying data segments corresponding respectively to a first group of objects;

submitting the identifying data segments to a first interface associated with an application program that maintains the first object store, wherein the first object store includes a second group of objects that potentially includes at least some of the first group of objects;

in response to submitting said identifying data segments, receiving a first list of handles, said handles corresponding respectively to the objects of the first group;

querying a second interface associated with the application program for a second list of handles, said handles corresponding respectively to the objects of the second group;

matching any handles of the first and second lists that correspond to the same object; and querying a third interface associated with the application program with matching handles from the first and second lists to determine if the object to which they correspond has been changed in the first object store.

15. One or more computer-readable media as recited in claim 14, wherein said matching comprises querying a fourth interface associated with the application program to determine whether handles of the first and second lists correspond to the same object.

16. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform concluding that a particular object has been deleted from the first object store if the handle from the first list corresponding to said particular object does not have a matching handle from the second list.

17. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform concluding that a particular object has been added to the first object store if the handle from the second list corresponding said particular object does not have a matching handle from the first list.

18. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform updating the second object store with any objects that have been changed in the first object store.

19. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform:

receiving a modified form of a particular object from the second object store;

identifying which handle from the second list corresponds to said particular object;

providing the identified handle and the modified form of said particular object to a fourth interface associated with the application program; and

instructing the application program to replace the object corresponding to the identified handle with the modified form of the object.

20. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform:

maintaining a mapping between handles and object identifiers that are used in the second store to identify objects;

receiving a modified form of a particular object from the second object store;

receiving an object identifier corresponding to said particular object;

referencing the mapping with the object identifier to identify which handle corresponds to said particular object;

providing the identified handle and the modified form of said particular object to a fourth interface associated with the application program; and

instructing the application program to replace the object corresponding to the identified handle with the modified form of the object.

21. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform:

receiving a modified form of a particular object from the second object store;

identifying which handle from the second list corresponds to said particular object;

providing the identified handle and the modified form of said particular object to a fourth interface associated with the application program;

instructing the application program to replace the object corresponding to the identified handle with the modified form of the particular object;

requesting new identifying data from a fifth interface associated with the application program for the object corresponding to the identified handle; and

updating the reference store with the new identifying data.

22. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform:

requesting new identifying data for a particular object from a fourth interface associated with the application program if the particular object has been changed in the first object store; and

updating the reference store with the new identifying data for said particular object.

23. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform maintaining a mapping between handles and object identifiers that are used in the second store to identify objects.

24. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed, perform:

maintaining a mapping between handles and object identifiers that are used in the second store to identify objects; and

maintaining a mapping between the identifying data segments and the object identifiers.

25. One or more computer-readable media as recited in claim 14, further comprising computer executable instructions that, when executed when a particular object has been changed in the first object store, perform:

receiving a modified form of the particular object from the second object store;

identifying which handle from the second list corresponds to said particular object;

providing the identified handle and the modified form of said particular object to a fourth interface associated with the application program;

querying a fifth interface associated with the application program for conflict resolution text; and

prompting a user using the conflict resolution text.

26. A set of application program interfaces embodied on a computer-readable medium for execution on a computer in conjunction with an application program that maintains an object store, comprising:

a first interface that receives an identifying data segment and that returns a handle corresponding to the identifying data segment; and

a second interface that receives the handle and in response returns an indication of whether an object corresponding to the identifying data segment has been changed in the object store.

27. A set of application program interfaces as recited in claim 26, wherein the second interface also receives a second handle, wherein the second interface compares identifying data segments corresponding to the received handles to determine whether the object has been changed in the object store.

28. A set of application program interfaces as recited in claim 26, further comprising a third interface that enumerates a list of handles corresponding respectively to objects in the object store.

29. A set of application program interfaces as recited in claim 26, further comprising:

a third interface that enumerates a list of handles corresponding respectively to objects in the object store;

wherein the second interface also receives a second handle from the list of handles; and

wherein the second interface compares identifying data segments corresponding to the received handles to determine whether the object has been changed in the object store.

30. A set of application program interfaces as recited in claim 26, further comprising a third interface that receives two handles and that returns an indication of whether the two handles correspond to the same object.

31. A set of application program interfaces as recited in claim 26, further comprising:

a third interface that receives an object having individual properties and that stores at least some of the individual properties in the object store; and

a fourth interface that returns data representing properties of the object that are not supported by the object store.

32. A set of application program interfaces embodied on a computer-readable medium for execution on a computer in conjunction with an application program that maintains an object store, comprising:

a first interface that receives an identifying data segment and that returns a handle corresponding to the identifying data segment;

a second interface that enumerates a list of handles corresponding respectively to objects in the object store and to identifying data segments from the respective objects; and

a third interface that receives two handles and in response compares the identifying data segments corresponding to the handles and returns an indication of whether the handles represent an unchanged object.

33. A set of application program interfaces as recited in claim 32, further comprising a fourth interface that receives two handles and that returns an indication of whether the two handles correspond to the same object.

34. A set of application program interfaces as recited in claim 32, further comprising:

a fourth interface that receives an object having individual properties and that stores at least some of the individual properties in the object store; and

a fifth interface that returns data representing properties of the object that are not supported by the object store.

35. A set of application program interfaces as recited in claim 32, further comprising:

a fourth interface that receives two handles and that returns an indication of whether the two handles correspond to the same object;

a fifth interface that receives an object having individual properties and that stores at least some of the individual properties in the object store; and

a sixth interface that returns data representing properties of the object that are not supported by the object store.

36. A method for synchronizing objects between first and second object stores, wherein the second object store has objects that include properties not supported by the first object store, comprising:

sending an object from the second object store to an interface associated with an application program that maintains the first object store;

storing at least some individual properties of the object in the first object store;

returning data representing unsupported individual properties without storing them in the first object store; and

storing the returned data in a reference store that is not implemented by the application program.

37. A method as recited in claim 36, further comprising returning the object, including the unsupported individual properties, to the second object store.

38. A method as recited in claim 36, further comprising:

sending the returned data from the reference store to an interface associated with the application program; and

returning the object, including the unsupported individual properties, to the second object store.

39. A system for synchronizing objects between first and second object stores, wherein the second object store has objects that include properties not supported by the first object store, comprising:

a primary computer;

an application program that executes on the primary computer to maintain the first object store;

a synchronization manager that executes on the primary computer;

the synchronization manager being configured to send an object from the second store to an interface associated with the application program;

the application program and its interfaces being configured to store at least some individual properties of the object in the first object store and to return data representing unsupported individual properties to the synchronization manager without storing them in the first object store; and

the synchronization manager being further configured to store the returned data in a reference store.

40. A system as recited in claim 39, wherein:

the synchronization manager is further configured to request the object from an interface associated with the application program;

wherein the application program and its interfaces receive the data representing the unsupported properties from the synchronization manager and return the object to the synchronization manager; and

wherein the synchronization manager sends the object to the second object store.

41. A system as recited in claim 39, further comprising a portable information device that maintains the second object store.